**Node.js Basics: Exploring Core Concepts** - 100 points possible

**Objective**: To learn the fundamental concepts and architecture of Node.js, including its asynchronous nature and module system, and to develop the practical skill of effectively using online documentation.

**Instructions**: Answer the questions below in your own words, using the links provided as a starting point. **Please do some reading and try to learn something, don't just speed through it.** You may need to access other sources than the supplied links to answer the questions provided.

Create a **nodeJS** folder in your root directory. Download this document (File>Download Choose Microsoft Word). Fill out the answers to the questions below and commit it to the **nodeJS** folder.

**Questions**:

1. Understanding Node.js:

What is Node.js, and what is Node.js unique advantage?

It is an open-source and cross-platform JS habitat. It is a well known tool for any kind of project! It runs in a single process, and it has a performance advantage because it runs the JS engine outside of the browser.

<https://nodejs.org/en/learn/getting-started/introduction-to-nodejs>

1. Modules in Node.js:

Node.js historically used CommonJS (require). What is the modern module system now supported by Node.js, and what keyword does it use?

The modern module system now, is ESM. It’s keyword is import.

<https://nodejs.org/en/learn/modules/modules-introduction>

1. The Event Loop:

What is the Node.js event loop, and how does its operation enable Node.js to perform fast, non-blocking I/O operations?

It allows Node.js to perform fast and non-blocking I/O operations. It does this by initializing the event loop, then processes the input script, then it begins processing the event loop.

<https://nodejs.org/en/learn/asynchronous-work/event-loop-timers-and-nexttick>

1. Working with Binary Data:

What is a "Buffer" in Node.js, and what type of data does it handle?

It is used to manage and store binary data.

<https://www.geeksforgeeks.org/node-js/what-is-buffer-in-node-js/>

1. “fs" module in Node.js

What does the "fs" module in Node.js allow you to do?

It provides an API for tinkering with the file system.

<https://www.geeksforgeeks.org/node-js/node-js-file-system/>

1. Creating a Web Server:

Explain the basic steps of building a web server using node.js and the http module.

Create a JavaScript and a json file. Run index.js using the command “node index.js”.

[https://www.geeksforgeeks.org/node-js/how-to-build-a-simple-web-server-with-node-js/#using-builtin-http-module](https://www.geeksforgeeks.org/node-js/how-to-build-a-simple-web-server-with-node-js/" \l "using-builtin-http-module)

1. Managing Packages with npm:

What is "npm," and what does it help you manage?

It is the planet’s biggest software registry. It manages private development projects.

<https://docs.npmjs.com/about-npm>

8. Package.json

What is the package.json file?

It is the heart of the Node.js sytem because it contains the metadata of the project.

<https://www.geeksforgeeks.org/node-js/node-js-package-json/>

9. Asynchronous Control Flow (Promises & Async/Await)

Modern Node.js relies heavily on asynchronous code to prevent blocking the server.

How do Promises and the async/await keywords help developers write cleaner asynchronous code compared to traditional callbacks?

Helps make code cleaner, easier to read, and easier to debug.

<https://nodejs.org/en/learn/asynchronous-work/javascript-asynchronous-programming-and-callbacks>

[https://nodejs.org/en/learn/asynchronous-work/discover-promises-in-nodejs#using-asyncawait-with-promises](https://nodejs.org/en/learn/asynchronous-work/discover-promises-in-nodejs" \l "using-asyncawait-with-promises)

10. The process Global Object

Node.js provides a global object called **process** that provides information about the current Node.js process.

How can you use process.env or process.argv to access environment variables or command-line arguments? A command is able to load all environmental variables from the .env to the process.env.

<https://nodejs.org/en/learn/command-line/how-to-read-environment-variables-from-nodejs>